NYCU Pattern Recognition, Homework 1

Deadline: March 23, 23:59

Part. 1, Coding (60%):

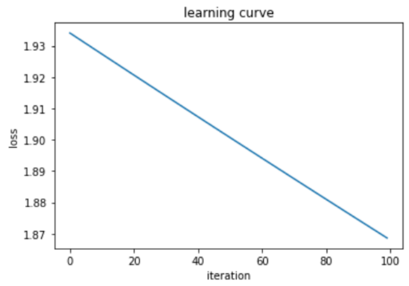
In this coding assignment, you need to implement linear regression by using only NumPy, then train your implemented model using **Gradient Descent** by the provided dataset and test the performance with testing data. Find the sample code and data on the GitHub page https://github.com/NCTU-VRDL/CS_AT0828/tree/main/HW1

We suggest using the hyper-parameters below:

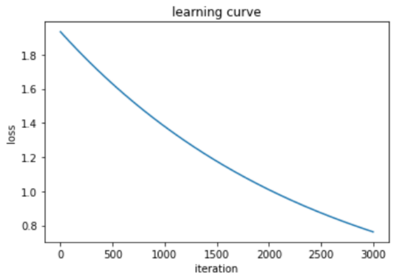
- Loss function: Mean Square Error
- Learning rate: 1e-4
- Number of training iteration: 100

Please note that only NumPy can be used to implement your model, you will get no points by simply calling sklearn.linear_model.LinearRegression. Moreover, please train your regression model using Gradient Descent, not the closed-form solution.

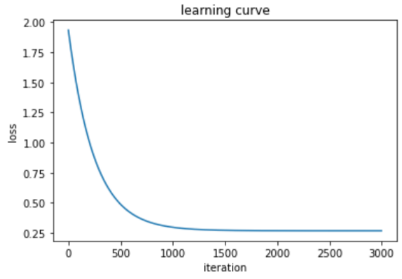
1. (15%) Plot the learning curve of the training, you should find that loss decreases after a few iterations (x-axis=iteration, y-axis=loss, Matplotlib or other plot tools is available to use)



Case 1. Learning rate = 1e-4, iteration = 100



Case 2. Learning rate = 1e-4, iteration = 3000



Case 3. Learning rate = 1e-3, iteration = 3000

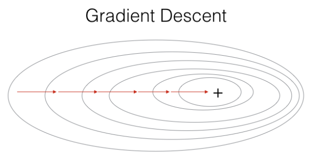
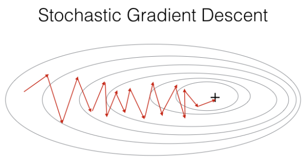
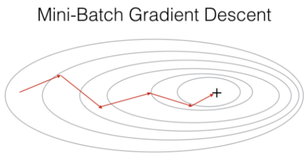
2. (15%) What's the Mean Square Error of your prediction and ground truth (prediction=model(x_test), ground truth=y_test)

	learning rate	iteration	MSE
Case 1	1e-4	100	0.1856559177908737
Case 2	1e-4	3000	0.0990770112570738
Case 3	1e-3	3000	0.06863312002230136

3. (15%) What're the weights and intercepts of your linear model?

	learning rate	iteration	B0 (intercept)	B1 (weight)
Case 1	1e-4	100	1.030718158238762	1.138474746257781
Case 2	1e-4	3000	0.9201768030405354	0.9981224425147557
Case 3	1e-3	3000	0.7850795467240413	0.8188173839216217

4. (10%) What's the difference between Gradient Descent, Mini-Batch Gradient Descent, and Stochastic Gradient Descent?

	Gradient Descent	Stochastic Gradient Descent	Mini-Batch Gradient Descent
diagram	 Gradient Descent	 Stochastic Gradient Descent	 Mini-Batch Gradient Descent
dataset	not suggested for huge training samples	large training samples	not suggested for huge training samples
batch	using the whole training sample	using a single training sample	using a fixed number of training samples
learning rate	fixed	randomly updated	fixed
speed	low	high (less computationally expensive)	medium
accuracy	high (gives optimal solution)	low (gives good solution but not optimal)	medium

5. (5%) All your codes should follow the PEP8 coding style and with clear comments

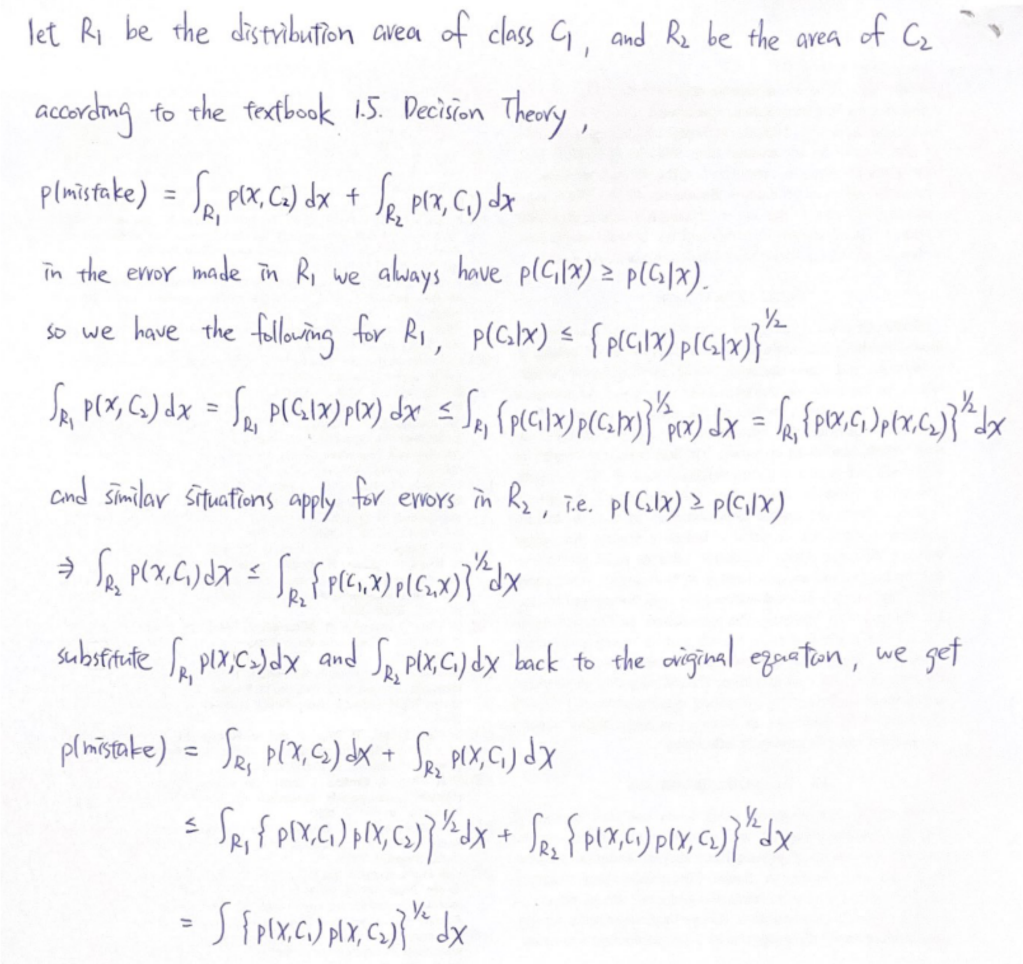
Part. 2, Questions (40%):

1. (10%) Suppose that we have three colored boxes R (red), B (blue), and G (green). Box R contains 3 apples, 4 oranges, and 3 guavas, box B contains 2 apples, 0 orange, and 2 guavas, and box G contains 12 apples, 4 oranges, and 4 guavas. If a box is chosen at random with probabilities p(R)=0.2, p(B)=0.4, p(G)=0.4, and a piece of fruit is removed from the box (with equal probability of selecting any of the items in the box), then what is the probability of selecting guava? If we observe that the selected fruit is in fact an apple, what is the probability that it came from the blue box?
    - the probability of selecting guava:
      0.2*(3/10) + 0.4*(2/4) + 0.4*(4/20) = 0.34
    - the probability that the selected apple came from the blue box:
      0.4*(2/4) / (0.2*(3/10) + 0.4*(2/4) + 0.4*(12/20)) = 0.4

2. (15%) Consider two nonnegative numbers a and b, and show that, if $a \le b$, then $a \le (ab)^{1/2}$. Use this result to show that, if the decision regions of a two-class classification problem are chosen to minimize the probability of misclassification, this probability will satisfy

$$p(mistake) \le \int \{p(x, C_1)\, p(x, C_2)\}^{1/2}\, dx\,.$$

(Hint: Please refer to the textbook 1.5. Decision Theory)



let R_1 be the distribution area of class C_1, and R_2 be the area of C_2

according to the textbook 1.5. Decision Theory,

$$p(\text{mistake}) = \int_{R_1} p(x, C_2)\, dx + \int_{R_2} p(x, C_1)\, dx$$

in the error made in R_1 we always have $p(C_1|x) \ge p(C_2|x)$.

so we have the following for R_1, $p(C_2|x) \le \{p(C_1|x)\, p(C_2|x)\}^{1/2}$

$$\int_{R_1} p(x, C_2)\, dx = \int_{R_1} p(C_2|x)\, p(x)\, dx \le \int_{R_1} \{p(C_1|x)\, p(C_2|x)\}^{1/2} p(x)\, dx = \int_{R_1} \{p(x, C_1)\, p(x, C_2)\}^{1/2} dx$$

and similar situations apply for errors in R_2, i.e. $p(C_2|x) \ge p(C_1|x)$

$$\Rightarrow \int_{R_2} p(x, C_1)\, dx \le \int_{R_2} \{p(C_1, x)\, p(C_2, x)\}^{1/2} dx$$

substitute $\int_{R_1} p(x, C_2)\, dx$ and $\int_{R_2} p(x, C_1)\, dx$ back to the original equation, we get

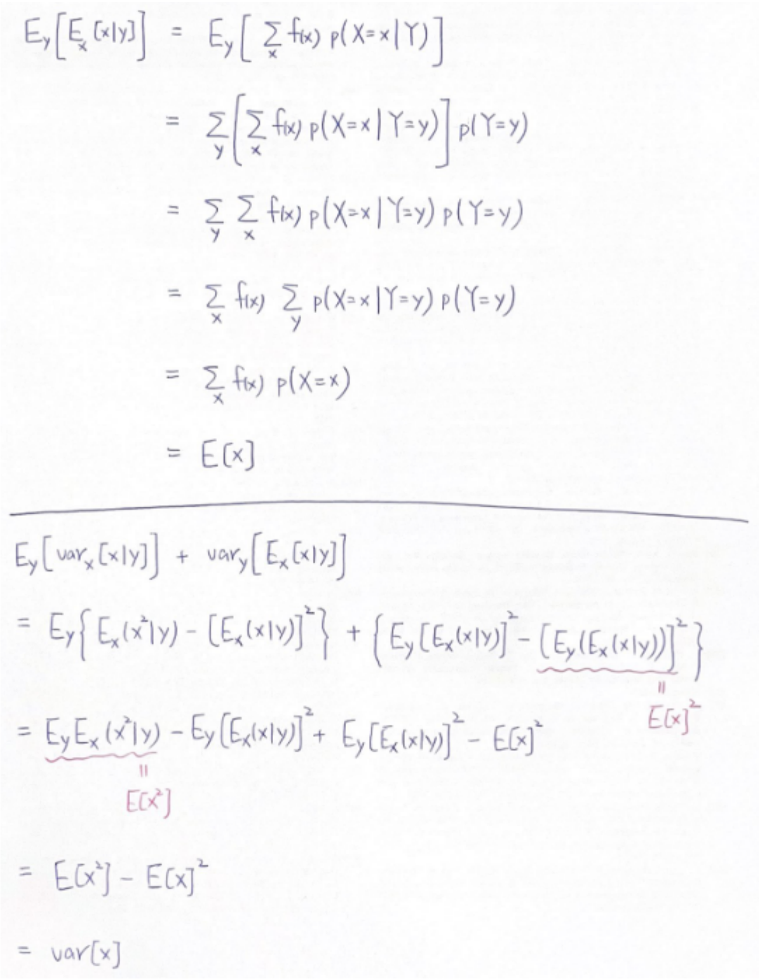
$$\begin{aligned} p(\text{mistake}) &= \int_{R_1} p(x, C_2)\, dx + \int_{R_2} p(x, C_1)\, dx \\ &\le \int_{R_1} \{p(x, C_1)\, p(x, C_2)\}^{1/2} dx + \int_{R_2} \{p(x, C_1)\, p(x, C_2)\}^{1/2} dx \\ &= \int \{p(x, C_1)\, p(x, C_2)\}^{1/2} dx \end{aligned}$$

3. (15%) Consider two variables x and y with joint distribution $p(x, y)$. Prove the following two results

$$E[x] = E_y\left[E_x[x|y]\right]$$

$$var[x] = E_y\left[var_x[x|y]\right] + var_y\left[E_x[x|y]\right].$$

Here $E_x[x|y]$ denotes the expectation of x under the conditional distribution $p(x|y)$, with a similar notation for the conditional variance.



$$\begin{aligned} E_y\left[E_x[x|y]\right] &= E_y\Big[\sum_x f(x)\, p(X=x|Y)\Big] \\ &= \sum_y \Big[\sum_x f(x)\, p(X=x|Y=y)\Big]\, p(Y=y) \\ &= \sum_y \sum_x f(x)\, p(X=x|Y=y)\, p(Y=y) \\ &= \sum_x f(x) \sum_y p(X=x|Y=y)\, p(Y=y) \\ &= \sum_x f(x)\, p(X=x) \\ &= E[x] \end{aligned}$$

$$E_y\left[var_x[x|y]\right] + var_y\left[E_x[x|y]\right]$$

$$= E_y\left\{E_x(x^2|y) - [E_x(x|y)]^2\right\} + \left\{E_y[E_x(x|y)]^2 - \underbrace{[E_y(E_x(x|y))]^2}_{\substack{\shortparallel \\ E[x]^2}}\right\}$$

$$= \underbrace{E_yE_x(x^2|y)}_{\substack{\shortparallel \\ E[x^2]}} - E_y[E_x(x|y)]^2 + E_y[E_x(x|y)]^2 - E[x]^2$$

$$= E[x^2] - E[x]^2$$

$$= var[x]$$